資工碩一 60247031S 朱聖池 DataMining\_FinalProject

1. 我選擇Topic 3來實作貝氏分類法，並且選擇下列資料集：

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **Acute Inflammations Data Set**  **Abstract**: The data was created by a medical expert as a data set to test the expert system, which will perform the presumptive diagnosis of two diseases of the urinary system. | http://archive.ics.uci.edu/ml/assets/MLimages/Large184.jpg |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **Data Set Characteristics:** | Multivariate | **Number of Instances:** | 120 | **Area:** | Life | | **Attribute Characteristics:** | Categorical, Integer | **Number of Attributes:** | 6 | **Date Donated** | 2009-02-11 | | **Associated Tasks:** | Classification | **Missing Values?** | No | **Number of Web Hits:** | 46849 | |

Attribute Information:

a1 Temperature of patient { 35C-42C }

a2 Occurrence of nausea { yes, no }

a3 Lumbar pain { yes, no }

a4 Urine pushing (continuous need for urination) { yes, no }

a5 Micturition pains { yes, no }

a6 Burning of urethra, itch, swelling of urethra outlet { yes, no }

由a1~a6的屬性來決定下列d1,d2的值，而d1,d2是沒有關係的：

d1 decision: Inflammation of urinary bladder { yes, no } //Class 1

d2 decision: Nephritis of renal pelvis origin { yes, no } //Class 2

程式：

Algo:

Begin

1. 檔案讀入陣列
2. 從120筆資料隨機取80筆作為Traning Data
3. 算在P( Rule|D1 = Y),P( Rule|D1 = N) ,P( Rule|D2 = Y),P( Rule|D2 = N)下發生機率

存至D\_probability陣列

1. 因為a1屬性是Contingous算mean及varience存至a1\_init陣列
2. 從120筆資料隨機取40筆作為Test Data
3. 計算其所有Rule之機率，算其Class存至testDecision陣列裡面
4. 開始比對TestData及Actually的Class算其正確率、精確值、回傳值、錯誤值、F-measure

End

主要是求：

P(D1 = Yes|Ruleset) => P(a1|D1=Yes)\* P(a2|D1=Yes)\* P(a3|D1=Yes)\* P(a4|D1=Yes)\* P(a5|D1=Yes)\*

P(a6|D1=Yes)

P(D1 = No|Ruleset) => P(a1|D1= No)\* P(a2|D1= No)\* P(a3|D1= No)\* P(a4|D1= No)\* P(a5|D1= No)\*

P(a6|D1= No)

P(D2 = Yes|Ruleset) => P(a1| D2 =Yes)\* P(a2| D2 =Yes)\* P(a3| D2 =Yes)\* P(a4| D2 =Yes)\* P(a5| D2 =Yes)\*

P(a6|D1=Yes)

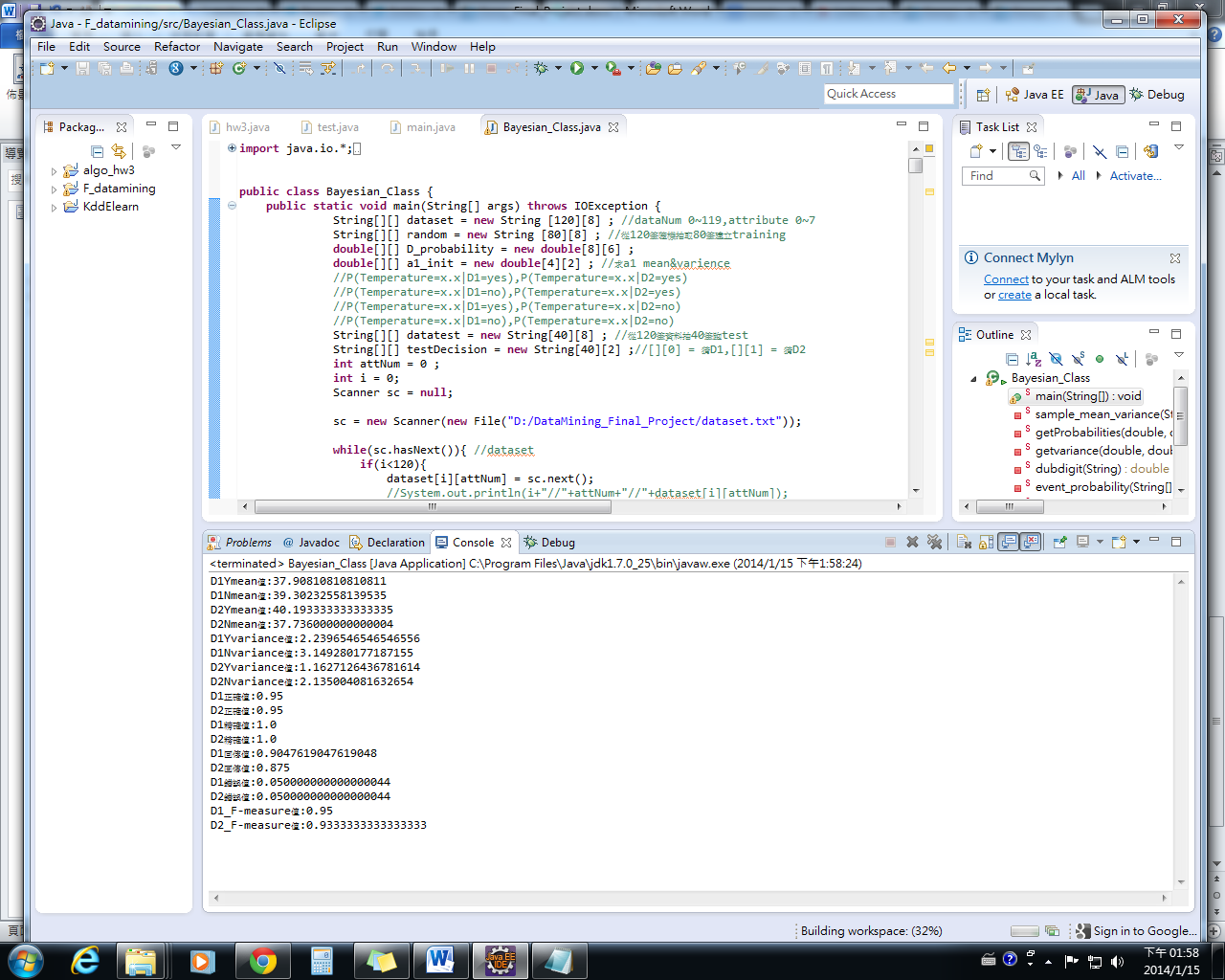
P(D2 = No|Ruleset) => P(a1| D2 = No)\* P(a2| D2 = No)\* P(a3| D2 = No)\* P(a4| D2 = No)\* P(a5| D2 = No)\*

P(a6| D2 = No)

然後看誰比較大而決定D1,D2。

程式執行：

因為是以eclipse編輯Java，需要先安裝JDK，dataser.txt檔案讀入需要在以下路徑才讀取不會錯誤並得到齊資料(D:/DataMining\_Final\_Project/dataset.txt")，然後以eclipse載入專案F\_datamining開始執行，會得以下結果。



選擇貝氏是因為比Decsion Tree好做，因為Decsion Tree在每次選擇一屬性要把資料split，然後才算其Gini值，主要把資料split有點麻煩。

一開始從120筆隨機不重複80筆作為TraningData，訓練出的model用40筆(從120筆隨機不重複選出)來測試其各種值，發現正確率都在9成以上。而a1~a6來決定d1和d2的class。

以下是D1依m不同時的數據值(五次加總後平均)

M=0~5時數據，D2(五次加總平均後)

Method for performance Evaluation：

1. Holdout:取一部分當traindata，剩下data當作testdata。但如果取出的資料可能不具代表性。
2. Random Subsampling:重複做Holdout，將正確率平均加總。而不確定要設定做幾次
3. Cross Validation: 每次取一部分的data當training data，剩下的當testing data，交叉驗證做直到每個data都有當過training data。
4. Bootstrap: 取一部分的data當training data然後放回，接著再取一部分的data當testing data

Holdout適合樣本數目多的，而Bootstrap適合樣本數目較少的